



Praxis (Systemprogrammierung)

1. **Signale:** Schreiben Sie ein C-Programm, das nach dem Start sofort `fork()` t. Der Sohnprozess soll in regelmäßigen Abständen ein Sternchen auf die Konsole schreiben – damit die Ausgabe direkt nach jedem Zeichen sichtbar wird, müssen Sie die Standardausgabe `stdout` „flushen“:

```
printf ("*"); fflush (stdout);
```

Nach jeder Ausgabe lassen Sie den Prozess 150.000 μ s warten, dazu nutzen Sie die Funktion `usleep()`. Der Vaterprozess soll alle 450.000 μ s ein Signal an seinen Sohn schicken und ihn dabei abwechselnd anhalten bzw. wieder fortsetzen. Das geht mit der Funktion `kill()`. Die benötigten Signale finden Sie in der Shell mit `kill -l`. Damit Sie das Programm übersetzen können, müssen Sie zwei Header-Dateien einbinden: Setzen Sie dazu an den Anfang des Programms die Zeilen

```
#include <stdio.h>  
#include <unistd.h>
```

Testen Sie Ihr Programm. Es sollte im Wechsel drei Sternchen ausgeben, dann genau solange warten, dann wieder drei Sternchen ausgeben etc.

2. **Signale:** Schreiben Sie ein C-Programm, das einen Signal-Handler für das Signal `SIG_USR1` einrichtet. Das Hauptprogramm soll einmal pro Sekunde einen global gespeicherten String `s` ausgeben, den Sie auf „#“ initialisieren. Jedes Mal, wenn das Signal `SIG_USR1` empfangen wird, ändern Sie diesen String in „A“ und beim nächsten Mal wieder zurück in „#“. Wenn Sie das Programm laufen lassen und einige Male das Signal `SIG_USR1` an den Prozess schicken (z. B. mit `killall -USR1 prozessname`), erhalten Sie also eine Ausgabe, die wie folgt aussieht:

```
#####AAAAAAAAA#####AAAA###...    (Wechsel nach jedem Signalempfang)
```

Praxis (IT-Sicherheit)

3. **C-time:** Laden Sie das Paket `stroke-0.1.3.tar.gz` von der Downloadseite herunter (<http://sourceforge.net/projects/stroke/files/stroke/stroke-0.1/> – auch über <http://stroke.sourceforge.net/> erreichbar), entpacken Sie das Paket und installieren Sie das Programm, indem Sie mit Root-Rechten

```
./configure  
make  
make install
```

eingeben. Danach steht Ihnen das Kommando `stroke` zur Verfügung, das eine leistungsfähigere Alternative zu `touch` ist. U. a. kann `stroke` die C-time (inode **change time**) ändern:

Geben Sie mit `stat /etc/fstab` alle Zeitangaben (A-time, M-time, C-time) der Datei `/etc/fstab` aus und ändern Sie mit `stroke` das Datum der C-time auf den 01.01.1970; das geht mit dem Kommando

```
stroke cY=1970,cM=1,cD=1 /etc/fstab
```

Überprüfen Sie das Ergebnis durch erneuten Aufruf von `stat`.